



DECLARATION UNDER 37 C.F.R. § 1.132
Application Serial No. 10/550,463
Attorney Docket No. 09877.0363

Customer No. 22,852
Attorney Docket No. **09877.0363**

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Application of:)
)
Alfredo GAMBIRASIO, <i>et al.</i>)
)
Serial No.: 10/550,463) Group Art Unit: 2874
)
Filed: September 26, 2005) Examiner: Omar R. Rojas
)
For: MICROSTRUCTURED OPTICAL) Confirmation No. 4626
FIBRE)
)

Commissioner for Patents
P.O. Box 1450
Alexandria, Virginia 22313-1450

Sir:

DECLARATION UNDER 37 C.F.R. § 1.132

EXHIBIT A

Block-iterative frequency-domain methods for Maxwell's equations in a planewave basis

Steven G. Johnson and J. D. Joannopoulos

*Dept. of Physics and Center for Materials Science and Engineering,
Massachusetts Institute of Technology, Cambridge, MA 02139 USA*

*stevenj@alum.mit.edu
<http://ab-initio.mit.edu/>*

Abstract: We describe a fully-vectorial, three-dimensional algorithm to compute the definite-frequency eigenstates of Maxwell's equations in arbitrary periodic dielectric structures, including systems with anisotropy (birefringence) or magnetic materials, using preconditioned block-iterative eigensolvers in a planewave basis. Favorable scaling with the system size and the number of computed bands is exhibited. We propose a new effective dielectric tensor for anisotropic structures, and demonstrate that $O(\Delta x^2)$ convergence can be achieved even in systems with sharp material discontinuities. We show how it is possible to solve for interior eigenvalues, such as localized defect modes, without computing the many underlying eigenstates. Preconditioned conjugate-gradient Rayleigh-quotient minimization is compared with the Davidson method for eigensolution, and a number of iteration variants and preconditioners are characterized. Our implementation is freely available on the Web.

©2001 Optical Society of America

OCIS codes: (000.4430) Numerical approximation and analysis

References and links

1. See, e.g., J. D. Joannopoulos, P. R. Villeneuve, and S. Fan, "Photonic crystals: putting a new twist on light," *Nature (London)* **386**, 143-149 (1997).
2. S. G. Johnson and J. D. Joannopoulos, The MIT Photonic-Bands Package home page <http://ab-initio.mit.edu/mpb/>.
3. K. M. Ho, C. T. Chan, and C. M. Soukoulis, "Existence of a photonic gap in periodic dielectric structures," *Phys. Rev. Lett.* **65**, 3152-3155 (1990).
4. H. S. Sözüer and J. W. Haus, "Photonic bands: convergence problems with the plane-wave method," *Phys. Rev. B* **45**, 13962-13972 (1992).
5. R. D. Meade, A. M. Rappe, K. D. Brommer, J. D. Joannopoulos, and O. L. Alerhand, "Accurate theoretical analysis of photonic band-gap materials," *Phys. Rev. B* **48**, 8434-8437 (1993). Erratum: S. G. Johnson, *ibid* **55**, 15942 (1997).
6. T. Suzuki and P. K. L. Yu, "Method of projection operators for photonic band structures with perfectly conducting elements," *Phys. Rev. B* **57**, 2229-2241 (1998).
7. K. Busch and S. John, "Liquid-crystal photonic-band-gap materials: the tunable electromagnetic vacuum," *Phys. Rev. Lett.* **83**, 967-970 (1999).
8. J. Jin, *The Finite-Element Method in Electromagnetics* (Wiley, New York, 1993), Chap. 5.7.
9. A. Figotin, Y. A. Godin, "The computation of spectra of some 2D photonic crystals," *J. Comput. Phys.* **136**, 585-598 (1997).
10. W. C. Saylor, F. M. Mueller, and P. R. Villeneuve, "Augmented-plane-wave method for photonic band-gap materials," *Phys. Rev. B* **57**, 8819-8822 (1998).
11. W. Axmann and P. Kuchment, "An efficient finite element method for computing spectra of photonic and acoustic band-gap materials: I. Scalar case," *J. Comput. Phys.* **150**, 468-481 (1999).
12. D. C. Dobson, "An efficient method for band structure calculations in 2D photonic crystals," *J. Comput. Phys.* **149**, 363-376 (1999).
13. D. Mogilevtsev, T. A. Birks, and P. St. J. Russell, "Localized function method for modeling defect modes in 2D photonic crystals," *J. Lightwave Tech.* **17**, 2078-2081 (1999).

14. S. J. Cooke and B. Levush, "Eigenmode solution of 2-D and 3-D electromagnetic cavities containing absorbing materials using the Jacobi-Davidson algorithm," *J. Comput. Phys.* **157**, 350–370 (2000).
15. K. M. Leung, "Defect modes in photonic band structures: a Green's function approach using vector Wannier functions," *J. Opt. Soc. Am. B* **10**, 303–306 (1993).
16. J. P. Albert, C. Jouanin, D. Cassagne, and D. Bertho, "Generalized Wannier function method for photonic crystals," *Phys. Rev. B* **61**, 4381–4384 (2000).
17. E. Lidorikis, M. M. Sigalas, E. N. Economou, and C. M. Soukoulis, "Tight-binding parameterization for photonic band-gap materials," *Phys. Rev. Lett.* **81**, 1405–1408 (1998).
18. See, e.g., K. S. Kunz and R. J. Luebbers, *The Finite Difference Time Domain Methods* (CRC, Boca Raton, Fla., 1993).
19. C. T. Chan, S. Datta, Q. L. Yu, M. Sigalas, K. M. Ho, C. M. Soukoulis, "New structures and algorithms for photonic band gaps," *Physica A* **211**, 411–419 (1994).
20. C. T. Chan, Q. L. Lu, and K. M. Ho, "Order- N spectral method for electromagnetic waves," *Phys. Rev. B* **51**, 16635–16642 (1995).
21. S. Fan, P. R. Villeneuve, and J. D. Joannopoulos, "Large omnidirectional band gaps in metallo-dielectric photonic crystals," *Phys. Rev. B* **54**, 11245–11251 (1996).
22. K. Sakoda and H. Shiroma, "Numerical method for localized defect modes in photonic lattices," *Phys. Rev. B* **56**, 4830–4835 (1997).
23. J. Arriaga, A. J. Ward, and J. B. Pendry, "Order N photonic band structures for metals and other dispersive materials," *Phys. Rev. B* **59**, 1874–1877 (1999).
24. A. J. Ward and J. B. Pendry, "A program for calculating photonic band structures, Green's functions and transmission/reflection coefficients using a non-orthogonal FDTD method," *Comput. Phys. Comm.* **128**, 590–621 (2000).
25. P. Yeh, *Optical Waves in Layered Media* (Wiley, New York, 1988).
26. J. B. Pendry and A. MacKinnon, "Calculation of photon dispersion relations," *Phys. Rev. Lett.* **69**, 2772–2775 (1992).
27. P. M. Bell, J. B. Pendry, L. M. Moreno, and A. J. Ward, "A program for calculating photonic band structures and transmission coefficients of complex structures," *Comput. Phys. Comm.* **85**, 306–322 (1995).
28. J. M. Elson and P. Tran, "Dispersion in photonic media and diffraction from gratings: a different modal expansion for the R-matrix propagation technique," *J. Opt. Soc. Am. A* **12**, 1765–1771 (1995).
29. J. M. Elson and P. Tran, "Coupled-mode calculation with the R-matrix propagator for the dispersion of surface waves on truncated photonic crystal," *Phys. Rev. B* **54**, 1711–1715 (1996).
30. J. Chongjun, Q. Bai, Y. Miao, and Q. Ruhu, "Two-dimensional photonic band structure in the chiral medium—transfer matrix method," *Opt. Commun.* **142**, 179–183 (1997).
31. V. A. Mandelshtam and H. S. Taylor, "Harmonic inversion of time signals," *J. Chem. Phys.* **107**, 6756–6769 (1997). Erratum: *ibid.*, **109**, 4128 (1998).
32. N. W. Ashcroft and N. D. Mermin, *Solid State Physics* (Holt Saunders, Philadelphia, 1976).
33. M. Frigo and S. G. Johnson, "FFTW: an adaptive software architecture for the FFT," in *Proc. 1998 IEEE Intl. Conf. on Acoustics, Speech, and Signal Processing* (Institute of Electrical and Electronics Engineers, New York, 1998), 1381–1384.
34. A. H. Stroud, *Approximate Calculation of Multiple Integrals* (Prentice-Hall, Englewood Cliffs, NJ, 1971).
35. J. Nadobny, D. Sullivan, P. Wust, M. Seebass, P. Deuffhard, and R. Felix, "A high-resolution interpolation at arbitrary interfaces for the FDTD method," *IEEE Trans. Microwave Theory Tech.* **46**, 1759–1766 (1998).
36. P. Yang, K. N. Liou, M. I. Mishchenko, and B.-C. Gao, "Efficient finite-difference time-domain scheme for light scattering by dielectric particles: application to aerosols," *Appl. Opt.* **39**, 3727–3737 (2000).
37. R. D. Meade, private communications.
38. M. C. Payne, M. P. Tater, D. C. Allan, T. A. Arias, and J. D. Joannopoulos, "Iterative minimization techniques for *ab initio* total-energy calculations: molecular dynamics and conjugate gradients," *Rev. Mod. Phys.* **64**, 1045–1097 (1992).
39. See, e.g., A. Edelman and S. T. Smith, "On conjugate gradient-like methods for eigen-like problems," *BIT* **36**, 494–509 (1996).
40. S. Ismail-Beigi and T. A. Arias, "New algebraic formulation of density functional calculation," *Comp. Phys. Commun.* **128**, 1–45 (2000).
41. E. R. Davidson, "The iterative calculation of a few of the lowest eigenvalues and corresponding eigenvectors of large real-symmetric matrices," *Comput. Phys.* **17**, 87–94 (1975).
42. M. Crouzeix, B. Philippe, and M. Sadkane, "The Davidson Method," *SIAM J. Sci. Comput.* **15**, 62–76 (1994).
43. G. L. G. Sleijpen and H. A. van der Vorst, "A Jacobi-Davidson iteration method for linear eigen-

- value problems," *SIAM J. Matrix Anal. Appl.* **17**, 401–425 (1996).
44. B. N. Parlett, *The Symmetric Eigenvalue Problem* (Prentice-Hall, Englewood Cliffs, NJ, 1980).
 45. H. A. van der Vorst, "Krylov subspace iteration," *Computing in Sci. and Eng.* **2**, 32–37 (2000).
 46. P. E. Gill, W. Murray, and M. H. Wright, *Practical Optimization* (Academic, London, 1981).
 47. J. J. Dongarra, J. Du Croz, I. S. Duff, and S. Hammarling, "A set of Level 3 Basic Linear Algebra Subprograms," *ACM Trans. Math. Soft.* **16**, 1–17 (1990).
 48. E. Anderson, Z. Bai, C. Bischof, S. Blackford, J. Demmel, J. Dongarra, J. Du Croz, A. Greenbaum, S. Hammarling, A. McKenney, and D. Sorensen, *LAPACK Users' Guide* (SIAM, Philadelphia, 1999).
 49. A. Edelman, T. A. Arias, and S. T. Smith, "The geometry of algorithms with orthogonality constraints," *SIAM J. Matrix Anal. Appl.* **20**, 303–353 (1998).
 50. A. H. Sameh and J. A. Wisniewski, "A trace minimization algorithm for the generalized eigenvalue problem," *SIAM J. Numer. Anal.* **19**, 1243–1259 (1982).
 51. B. Philippe, "An algorithm to improve nearly orthonormal sets of vectors on a vector processor," *SIAM J. Alg. Disc. Meth.* **8**, 396–403 (1987).
 52. J. J. Moré and D. J. Thuente, "Line search algorithms with guaranteed sufficient decrease," *ACM Trans. Math. Software* **20**, 286–307 (1994).
 53. S. Ismail-Beiji, private communications.
 54. P. R. Villeneuve, S. Fan, and J. D. Joannopoulos, "Microcavities in photonic crystals: mode symmetry, tunability, and coupling efficiency," *Phys. Rev. B* **54**, 7837–7842 (1996).
 55. L.-W. Wang and A. Zunger, "Solving Schrödinger's equation around a desired energy: application to Silicon quantum dots," *J. Chem. Phys.* **100**, 2394–2397 (1994).
-

1 Introduction

Optical systems have been the subject of enormous practical and theoretical interest in recent years, with a corresponding need for mathematical and computational tools. One fundamental approach in their analysis is eigenmode decomposition: the possible forms of electromagnetic propagation are expressed as a set of definite-frequency (time-harmonic) modes. In the absence of nonlinear effects, all optical phenomena can then be understood in terms of a superposition of these modes, and many forms of analytical study are possible once the modes are known. Of special interest are periodic (or translationally-symmetric) systems, such as photonic crystals (or waveguides), which give rise to many novel and interesting optical effects [1]. Another important basic system is that of resonant cavities, which confine light to a point-like region. There, the boundary conditions are, in principle, irrelevant if the mode is sufficiently confined—so they can be treated under the rubric of periodic structures as well via the "super-cell" technique. In this paper, we describe a fully-vectorial, three-dimensional method for computing general eigenmodes of arbitrary periodic dielectric systems, including anisotropy, based on the preconditioned block-iterative solution of Maxwell's equations in a planewave basis. A new effective dielectric tensor for anisotropic systems is introduced, and we also describe a technique for computing eigenvalues in the interior of the spectrum (e.g. defect modes) without computing the underlying bands. We present comparisons of different iterative solution schemes, preconditioners, and other aspects of frequency-domain calculations. The result of this work is available as a free and flexible computer program downloadable from the Web [2].

There are a few common approaches to eigen-decomposition of electromagnetic systems. One, which we employ in this paper, is to expand the fields as definite-frequency states in some truncation of a complete basis (e.g. planewaves with a finite cutoff) and then solve the resulting linear eigenproblem. Such methods have seen widespread use, with many variations distinguished by the critical choices of basis and eigensolver algorithm [3–17]. This "frequency-domain" method is discussed in further detail below. Another common technique involves the direct simulation of Maxwell's equations over time on a discrete grid by finite-difference time-domain (FDTD) algorithms [18]—one Fourier-transforms the time-varying response of the system to some input, and the

peaks in the resulting spectrum correspond to the eigenfrequencies [19–24]. (Care must be taken to ensure that the source is not accidentally near-orthogonal to an eigenmode.) This has the unique (and sometimes desirable) feature of finding the eigenfrequencies only—to compute the associated fields, one re-runs the simulation with a narrow-band filter for each state. (Time-domain calculations can also address problems of a dynamic nature, such as transmission processes, that are not so amenable to eigenmethods.) A third class of techniques are referred to as “transfer-matrix” methods: at a fixed frequency, one computes the transfer matrix relating field amplitudes at one end of a unit cell with those at the other end (via finite-difference, analytical, or other methods) [25–30]. This yields the transmission spectrum directly, and mode wavevectors via the eigenvalues of the matrix; in some sense, this is a hybrid of time- and frequency-domain. Transfer-matrix methods may be especially attractive when the structure is decomposable into a few more-easily solvable components, and also for other cases such as frequency-dependent dielectrics.

In any method, the computation is characterized by some number N of degrees of freedom (e.g., the number of planewaves or grid points), and one might be interested to compare how the number of operations (the “complexity”) in each algorithm scales with N . Unfortunately, there is no simple answer. As we shall see below, the complexity in frequency-domain is $O(i_c p N \log N) + O(i_c p^2 N)$ for a planewave basis, where $p \ll N$ is the number of desired eigenmodes and i_c is the number of iterations for the eigensolver to converge. In time-domain, the complexity is $O(i_t N)$ to find the frequencies, and $O(p i_t N)$ to also solve for the fields of p modes, where i_t is the number of time steps. (Transfer-matrix methods have too many variations to consider here.) The difficulty in both cases comes from the number of iterations, which scales in different ways depending upon how the problem size is increased. In frequency-domain, i_c is hard to predict, but we shall show below that it often grows only very slowly with p and N , so we can treat it as approximately constant (often < 20). In time-domain, however, i_t must increase linearly with the spatial resolution (a certain kind of N increase) to maintain stability [18]. It also increases inversely with the required frequency resolution, by the uncertainty principle of the Fourier transform: $i_t \Delta t \Delta f \sim 1$ (where Δt is the timestep). Not only, then, is i_t a large number (typically > 1000), but it must also increase dramatically to resolve closely-spaced modes (although this can be ameliorated somewhat by sophisticated signal processing [31]); in contrast, frequency-domain methods have no special problem resolving even degenerate modes. One traditional advantage of time-domain has been its ability to extract modes in the interior of the spectrum without computing the lower-frequency states, but we will show that this is feasible in frequency-domain as well. We feel that both time- and frequency-domain methods remain useful tools to extract eigenmodes from many structures.

There is sometimes concern that discontinuities in the dielectric function may cause poor convergence in a planewave basis. As is described in Sec. 2.3, however, this can be alleviated by the use of smoothed effective dielectric tensor [5], and we demonstrate that convergence proportional to the square of the spatial resolution, Δx^2 , can be achieved even for sharply discontinuous anisotropic dielectric structures.

In the paper that follows, we describe in greater detail our method for obtaining the eigenmodes of Maxwell’s equations, dividing the discussion into two parts: Maxwell’s equations and eigensolvers. First, we review how Maxwell’s equations can be cast as an eigenproblem for the frequencies, discuss the choice of basis and the computation of an effective dielectric tensor, and consider the critical selection of an approximate preconditioner. Second, we describe various block-iterative algorithms for solving this eigensystem; in principle this is independent of the equations being solved, but in practice there are a number of specific considerations. Throughout, we illustrate the methods

being compared with numerical results for example systems.

2 The Maxwell Eigenproblem

We first express Maxwell's equations as a linear eigenproblem, abstracting where possible from the differential equations in the individual field components to a higher-level view that better illuminates the overall process. To this end, we employ the Dirac notation of abstract operators \hat{A} and states $|H\rangle$ to provide a representation-independent expression for the fields and inner products, and later use ordinary matrix notation to indicate the transition to a finite problem. (The underlying equations remain fully vectorial.) This eigenproblem formulation has appeared in various forms elsewhere, and we begin by reviewing it here. The source-free Maxwell's equations for a linear dielectric $\varepsilon = \varepsilon(\vec{x})$ can be written in terms of only the magnetic field $|H\rangle$ [1]:

$$\vec{\nabla} \times \frac{1}{\varepsilon} \vec{\nabla} \times |H\rangle = -\frac{1}{c^2} \frac{\partial^2}{\partial t^2} |H\rangle, \quad (1)$$

$$\vec{\nabla} \cdot |H\rangle = 0. \quad (2)$$

We consider only states with definite frequency ω , i.e. a time-dependence $e^{-i\omega t}$. Furthermore, we suppose that the system is periodic—in that case, Bloch's theorem for periodic eigenproblems says that the states can be chosen to be of the form [32]:

$$|H\rangle = e^{i(\vec{k} \cdot \vec{x} - \omega t)} |H_{\vec{k}}\rangle, \quad (3)$$

where \vec{k} is the "Bloch wavevector" and $|H_{\vec{k}}\rangle$ is a *periodic* field (completely defined by its values in the unit cell). Eq. (1) then becomes the linear eigenproblem in the unit cell:

$$\hat{A}_{\vec{k}} |H_{\vec{k}}\rangle = (\omega/c)^2 |H_{\vec{k}}\rangle, \quad (4)$$

where $\hat{A}_{\vec{k}}$ is the positive semi-definite Hermitian operator:

$$\hat{A}_{\vec{k}} = \left(\vec{\nabla} + i\vec{k} \right) \times \frac{1}{\varepsilon} \left(\vec{\nabla} + i\vec{k} \right) \times. \quad (5)$$

All the familiar theorems of Hermitian eigenproblems apply. Because $|H_{\vec{k}}\rangle$ has compact support, the solutions are a discrete sequence of eigenfrequencies $\omega_n(\vec{k})$ forming a continuous "band structure" (or "dispersion relation") as a function of \vec{k} . These discrete bands (modes as a function of \vec{k}) provide a complete picture of all possible electromagnetic states of the system, but typically one is interested in only the lowest few. (For example, in a photonic crystal it is possible for there to be a band gap in the lower bands: a range of ω in which no states exist [1].) Furthermore, the modes at a given \vec{k} may be chosen to be orthonormal:

$$\left\langle H_{\vec{k}}^{(n)} \middle| H_{\vec{k}}^{(m)} \right\rangle = \delta_{n,m}, \quad (6)$$

where $\delta_{n,m}$ is the Kronecker delta.

2.1 The choice of basis

In frequency-domain methods, Eq. (4) is transformed into a finite problem by expanding the states in some truncated (possibly vectorial) basis $\{|b_m\rangle\}$:

$$|H_{\vec{k}}\rangle \cong \sum_{m=1}^N h_m |b_m\rangle. \quad (7)$$

This expression becomes exact as the number N of basis functions goes to infinity, assuming a complete basis. One then has the ordinary generalized eigenproblem

$$Ah = \left(\frac{\omega}{c}\right)^2 Bh, \quad (8)$$

where h is a column vector of the basis coefficients h_m , and A and B are $N \times N$ matrices with entries $A_{\ell m} = \langle b_\ell | \hat{A}_k | b_m \rangle$ and $B_{\ell m} = \langle b_\ell | b_m \rangle$. It is important to note that Eq. (8) by itself is incomplete, however—the modes must also satisfy the “transversality” constraint of Eq. (2); zero-frequency spurious modes are otherwise introduced, as can be seen by taking the divergence of both sides of Eq. (1).

In principle, one could compute the entries of A and B and then use a standard matrix algorithm to solve Eq. (8) directly, and this method is sometimes employed [3,4,6,7,10]. Such a computation, however, requires $O(N^2)$ storage for the matrices and $O(N^3)$ work for the diagonalization, making it impractical for large problems. Fortunately, since only p bands are typically desired, for some $p \ll N$, iterative methods are available to compute the bands with only $O(pN)$ storage and roughly $O(p^2N)$ work; these methods are the subject of Sec. 3. The relevant property of iterative methods is this: they require only a fast, ideally $O(N)$, method to compute the products Ah and Bh , with no need to store the matrices explicitly.

The choice of basis functions $|b_n\rangle$, then, is determined by three factors. First, they should form an compact representation so that a reasonable N yields good accuracy. Second, a convenient and efficient method for computing Ah and Bh must be available. Third, they should be inherently transverse, or otherwise provide an inexpensive way to maintain the constraint of Eq. (2).

2.1.1 The planewave basis

We chose to use a planewave basis (for reasons described below) [3–7], in which $|b_m\rangle = e^{i\vec{G}_m \cdot \vec{x}}$ for some reciprocal-lattice vectors \vec{G}_m ; the truncation N is determined by choosing a maximum cutoff for the magnitude of \vec{G}_m . Strictly speaking, a cutoff magnitude would result in a spherical volume of \vec{G} vectors, but we expand this into a parallelepiped volume so that the transformation between planewave and spatial representations takes the convenient form of a Discrete Fourier Transform (DFT). (Such an extension also removes an ambiguity of the order in which to invert and Fourier-transform ε [5].) The planewave set then has a duality with a spatial grid, which is often a more intuitive representation. In particular, suppose that the three primitive lattice vectors (the units of periodicity) are $\{\vec{R}_1, \vec{R}_2, \vec{R}_3\}$ and the primitive reciprocal-lattice vectors are $\{\vec{G}_1, \vec{G}_2, \vec{G}_3\}$, defined by $\vec{R}_i \cdot \vec{G}_j = 2\pi\delta_{i,j}$ [32]. Then the basis functions are $|b_{m_1, m_2, m_3}\rangle = e^{i\sum_j m_j \vec{G}_j \cdot \vec{x}}$ with $m_j = -\lfloor N_j/2 \rfloor + 1, \dots, \lfloor N_j/2 \rfloor$,¹ $N = N_1 N_2 N_3$, and Eq. (7) for the spatial field becomes:

$$\vec{H}_{\vec{k}} \left(\sum_k n_k \vec{R}_k / N_k \right) = \sum_{\{m_j\}} \vec{h}_{\{m_j\}} e^{i\sum_{j,k} m_j \vec{G}_j \cdot n_k \vec{R}_k / N_k} = \sum_{\{m_j\}} \vec{h}_{\{m_j\}} e^{2\pi i \sum_j m_j n_j / N_j}. \quad (9)$$

Here, $n_k = 0, \dots, N_k - 1$ describe spatial coordinates on an $N_1 \times N_2 \times N_3$ affine grid along the lattice directions. This is precisely a three-dimensional DFT, and can be computed by an efficient Fast Fourier Transform (FFT) algorithm [33] in $O(N \log N)$ time.

¹This is equivalent to $m_j = 0, \dots, N_j - 1$ for the DFT, in which m_j is interpreted modulo N_j , but choosing zero-centered wavevectors is important when taking derivatives of the basis.

Thus, in a planewave representation, the product Ah from Eq. (8) can be computed in $O(N \log N)$ time by taking the curl in wavevector space (just the cross-product with $\vec{k} + \vec{G}_m$), computing the FFT, multiplying by $\widetilde{\epsilon^{-1}}$, computing the inverse FFT, and taking the curl again [5]:

$$A\epsilon_m = -(\vec{k} + \vec{G}_\ell) \times \dots \text{IFFT} \dots \widetilde{\epsilon^{-1}} \dots \text{FFT} \dots (\vec{k} + \vec{G}_m) \times . \quad (10)$$

The determination of an effective inverse dielectric tensor, $\widetilde{\epsilon^{-1}}$, is discussed in Sec. 2.3. The matrix B is simply the identity, thanks to the orthonormality of the basis.

Since the basis functions themselves are scalars in this case, the amplitudes \tilde{h}_m in Eq. (7) must be vectors. In addition to Eq. (8), the field must satisfy the transversality constraint, and it is here that a key advantage of the planewave basis becomes apparent: Eq. (2) becomes merely a local constraint on the amplitudes, $\tilde{h}_m \cdot (\vec{k} + \vec{G}_m) = 0$. For each reciprocal vector \vec{G}_m one chooses a pair $\{\hat{u}_m, \hat{v}_m\}$ of orthonormal unit vectors that are perpendicular to $\vec{k} + \vec{G}_m$, and writes the amplitude as $\tilde{h}_m = h_m^{(1)} \hat{u}_m + h_m^{(2)} \hat{v}_m$. Then the basis is intrinsically transverse, and one can treat Eq. (8) as an ordinary eigenproblem of rank $n = 2N$ without worrying about any constraint.

2.1.2 Other possible bases

The planewave basis has at least two potential disadvantages: first, it corresponds to a uniform spatial grid, and may thus be a less economical representation than one based on e.g. a general mesh; second, the computation of the Maxwell operator A requires $O(N \log N)$ time instead of $O(N)$, although the difference may be small in practice. Both of these problems could be overcome by using a different basis—for example, a traditional finite-element basis formed of localized functions on an unstructured mesh. Such a basis, however, would make it more difficult to maintain the transversality constraint, which is why we eschew it in our implementation. (One way around this might be to replace the magnetic field with the vector potential, $\vec{H} = \vec{\nabla} \times \vec{A}$, although this introduces higher-order derivatives into the eigenproblem.) Alternatively, it is possible to solve the eigenproblem without transversality and *a posteriori* identify and remove the resulting spurious modes (which lie at zero frequency unless a “penalty” term is added to the eigen-equation) [8, 14].

In two dimensions, though, transversality ceases to be a problem: one simply chooses the magnetic field along z (TE fields) or the electric field along z (TM fields, for which the eigenproblem could be recast in \vec{E} or \vec{D}). This fact has been employed by various researchers to implement finite-element or other-basis frequency-domain methods in 2D [8–14]. (Our implementation also supports the 2D TE/TM case, of course.)

Given the eigenmodes for the primitive cell of a lattice, it may be possible to use them to construct a localized Wannier-function representation that is a useful basis for defect-mode calculations [15, 16], although the non-uniqueness of the Wannier functions must be resolved, e.g., by fitting to the precomputed band structure in “tight-binding” fashion [16]. Such a basis is automatically divergenceless since the constituent eigenmodes are transverse. Another possibility is a tight-binding basis that is not specified explicitly, but whose matrix elements are fitted to an existing band diagram [17].

2.2 Inversion symmetry

In general, the basis coefficients h are complex, but additional simplifications are possible when the dielectric function possesses inversion symmetry: $\epsilon(-\vec{x}) = \epsilon(\vec{x})$. The Fourier transform of a real and even function is real and even, so it follows that the

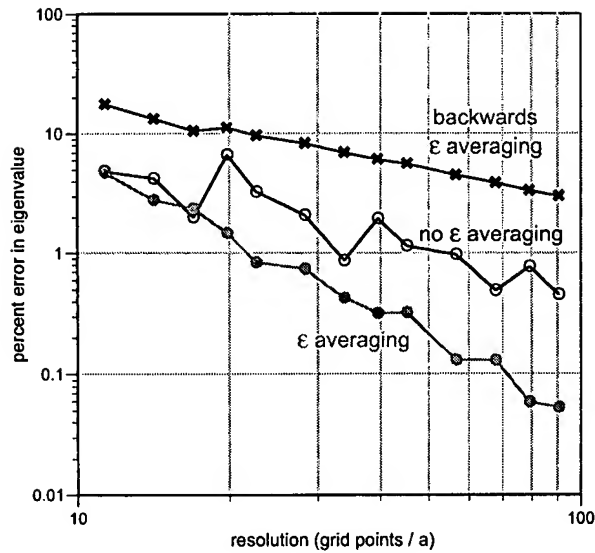


Fig. 1. Eigenvalue convergence as a function of grid resolution (grid points per lattice constant a) for three different methods of determining an effective dielectric tensor at each point: *no averaging*, simply taking the dielectric constant at each grid point; *averaging*, the smoothed effective dielectric tensor of Eq. (12); and *backwards averaging*, the same smoothed dielectric but with the averaging methods of the two polarizations reversed.

plane-wave representation of $\hat{A}_{\vec{k}}$, Eq. (10), is then a real-symmetric matrix. This means that the plane-wave amplitudes \tilde{h}_m can be chosen to be *purely real*, resulting in a factor of two savings in storage, more than a factor of two in time (due to more-efficient FFTs and matrix operations for real data), and possibly a reduction in the number of eigen-solver iterations (due to the reduced degrees of freedom). The spatial fields, in contrast, cannot generally be chosen as real—rather, with inversion symmetry they may be chosen to satisfy the property of the Fourier transform of a real function: $\tilde{H}_{\vec{k}}(\vec{x}) = \tilde{H}_{\vec{k}}(-\vec{x})^*$. Because inversion symmetry is extremely common in practical structures of interest, our implementation supports this optimization when the symmetry is present. For generality, however, we also handle the case of complex h for non-symmetric systems.

2.3 The effective dielectric tensor

When computing the product $\hat{A}_{\vec{k}} \left[\tilde{H}_{\vec{k}} \right]$ in a plane-wave basis, the multiplication by ε^{-1} is done in the spatial domain after a Fourier transform, so one might simply use the inverse of the actual dielectric constant at that point. Unfortunately, this can lead to suboptimal convergence of the frequencies as a function of N , due to the problems of representing discontinuities in a Fourier basis. It has been shown, however, that using a smoothed, *effective* dielectric tensor near dielectric interfaces can circumvent these problems, and achieve accurate results for moderate N [5]. In particular, near a dielectric interface, one must average the dielectric in two different ways according to effective-medium theory, depending upon the polarization of the incident light relative to the surface normal \hat{n} . For an electric field $\vec{E} \parallel \hat{n}$, one averages the inverse of ε ; for $\vec{E} \perp \hat{n}$, one takes the inverse of the average of ε . This results in an effective inverse dielectric tensor $\widetilde{\varepsilon^{-1}}$:

$$\widetilde{\varepsilon^{-1}} = \overline{\varepsilon^{-1}}P + \bar{\varepsilon}^{-1}(1 - P) \quad (11)$$

where P is the projection matrix onto \hat{n} : $P_{ij} = n_i n_j$. Here, the averaging is done over one voxel (cubic grid unit) around the given spatial point; if ϵ is constant, Eq. (11) reduces simply to ϵ^{-1} . (The original formulation was in terms of $\tilde{\epsilon}$, but is equivalent.)

We have generalized this procedure to handle the case of anisotropic (birefringent) dielectric materials, in which case the intrinsic ϵ is already a real-symmetric tensor. Or, more generally, for magnetic materials ϵ may be complex-Hermitian. In this case, an analogous equation is needed that: (i) produces a Hermitian effective inverse dielectric, (ii) reduces to Eq. (11) for the case of scalar ϵ , (iii) yields ϵ^{-1} for constant ϵ , (iv) retains the physical justification of the different averaging methods for the different polarizations, and (v) similarly improves convergence. The expression we propose that satisfies these criteria is:

$$\widetilde{\epsilon^{-1}} = \frac{1}{2} \left(\{ \overline{\epsilon^{-1}}, P \} + \{ \overline{\epsilon^{-1}}, (1 - P) \} \right), \quad (12)$$

where ϵ may be a tensor and $\{a, b\}$ denotes the anti-commutator $ab + ba$. Our first three conditions are manifestly satisfied. A physical justification of this formula is that a given averaging method should be used when either the field *or* the inverse ϵ times the field is in the appropriate direction relative to \hat{n} , hence the anti-commutators. To illustrate the convergence impact of Eq. (12), we consider a simple example case similar to that of [7]: an fcc lattice (lattice constant a) of close-packed spherical holes in dielectric ($\epsilon = 12$), where the holes are filled by an anisotropic “liquid crystal” material with an ϵ of 2 for fields along an “extraordinary” 011 (\hat{x}) direction and 1 otherwise. We compute the frequency of the ninth band (just below the gap for air holes) at the L point as a function of grid resolution for three cases: no ϵ averaging, with averaging as in Eq. (12), and with averaging *backwards* from Eq. (12) (P switched with $1 - P$). The results, shown in Fig. 1, exhibit a significant acceleration of convergence by the averaging; conversely, the poor convergence of the backwards averaging demonstrates the importance of polarization for the smoothing method. With the averaging of Eq. (12), we see that the error decreases with the square of the resolution, just as for standard FDTD methods [18].

As a practical matter, it can be cumbersome to compute (or even to define) the surface normal \hat{n} for complicated three-dimensional structures, so we implement an approximation. Given a flat dielectric interface, the normal direction is exactly $\int \vec{r} \epsilon$ over a spherical surface intersecting the interface, where \vec{r} is the vector from the center of the sphere. This procedure also yields the correct normal for spherical and cylindrical interfaces, by symmetry. Therefore, we use this spherical average to define the normal direction in all cases.² In order to compute the average, we employ a 50-point 11th-degree spherical-quadrature formula [34] to numerically integrate $\vec{r} \epsilon$ over a spherical surface inscribed within the ϵ averaging voxel. (Testing this method by computing the normal vectors to a large number of random planar surfaces, we found a mean angular error of about 5°.)

An interesting unanswered question is whether a similar effective ϵ tensor would improve the convergence of FDTD methods, for which scalar averaging methods have already been explored to improve modeling of discontinuous material interfaces [35, 36].

2.4 Preconditioners

A critical factor in the performance of an iterative eigensolver is the choice of “preconditioning” operator. As will be explained in Sec. 3, our preconditioner requires us to supply an *approximate* inverse \tilde{A}^{-1} of A such that $\tilde{A}^{-1}h$ can be computed quickly. This

²This method for defining \hat{n} can produce suboptimal results when the averaging voxel straddles two near-parallel dielectric interfaces. Preliminary investigations show that gains of a factor of two or more in eigenvalue accuracy are sometimes possible if a better approximation for \hat{n} is used in such cases.

choice of a “good” preconditioner is highly problem-dependent, and is unfortunately a matter of trial and error. We consider two possible preconditioners. The first is a diagonal preconditioner [5, 12, 37], inspired by the “kinetic energy” preconditioners often used in electronic calculations [38]. In this case, \tilde{A} is simply the diagonal elements of A :

$$\tilde{A}_{\ell m} = \left| \vec{k} + \vec{G}_m \right|^2 \delta_{\ell, m}, \quad (13)$$

where we have dropped an overall scaling factor (irrelevant to a preconditioner). The motivation for this approximation is that, for large \vec{G} components, the operator $\hat{A}_{\vec{k}}$ is dominated by the curl operations and not by the variations in ϵ . Computing $\tilde{A}^{-1}h$ is then an $O(N)$ diagonal operation.

We also consider a more accurate inverse [37]. Ideally, one would like to simply reverse each of the steps in computing Ah via the plane-wave representation of Eq. (10) to find an exact inverse. Note that a cross product of perpendicular vectors is invertible. So, the only operation of Eq. (10) that is not trivially reversible is the final cross product with $\vec{k} + \vec{G}_\ell$, since $-i\omega\vec{E} = \epsilon^{-1}\vec{\nabla} \times \vec{H}$ is not generally transverse (divergenceless). Since ϵ is normally piecewise constant and $\vec{\nabla} \cdot \epsilon\vec{E} = 0$, however, it is plausible that \vec{E} is “mostly” divergenceless. With this in mind, we approximate the operator $\hat{A}_{\vec{k}}$ by inserting projection operators \hat{P}_T onto the transverse field components:

$$\tilde{A} = \vec{\nabla} \times \hat{P}_T \frac{1}{\epsilon} \hat{P}_T \vec{\nabla} \times, \quad (14)$$

where the rightmost \hat{P}_T is superfluous and serves only to make the operator clearly Hermitian. Both curls are now reversible, since they act on transverse plane-waves. The inverse of Eq. (14) can be thus applied in $O(N \log N)$ time in a manner exactly analogous to Eq. (10): invert the $\vec{k} + \vec{G}_\ell$ cross product (projecting the result), FFT, multiply by ϵ , inverse FFT, and invert the $\vec{k} + \vec{G}_m$ cross product. This preconditioner is more expensive to compute than that of Eq. (13), but we shall show that it yields a substantial speedup in the eigensolver.

2.4.1 Removing the singularity

There is one evident obstacle in this preconditioning process—at $\vec{k} = 0$ (the “ Γ ” point), the operator $\hat{A}_{\vec{k}}$ is singular. (This will also be a problem for iterative methods such as conjugate-gradient, which are known to converge poorly for ill-conditioned matrices [46].) This difficulty is easily overcome, however, because the singular solutions are known analytically: they are the constant-field solutions at $\omega = 0$. So, at Γ we simply remove these singular solutions from the basis space (only considering plane-waves with $\vec{G}_m \neq 0$), and re-insert them after we have solved for the non-singular eigenvectors.

3 Iterative Eigensolvers

Iterative eigensolvers are fast methods for computing the $p \ll n$ smallest (or largest, or extremal) eigenvalues and eigenvectors of an $n \times n$ generalized eigenproblem $Ay = \lambda By$, by iteratively improving guesses for the eigenvectors until convergence is achieved. (B is the identity in our case, but we consider the full problem here for generality.) They are thus ideally suited for the problem of finding the few lowest eigenstates of Maxwell’s equations. Many iterative eigensolver algorithms have been developed, but we focus our investigations on two in particular: preconditioned conjugate-gradient minimization of the block Rayleigh quotient [5, 38–40], and the Davidson method [41–43] (an extension of Lanczos’s algorithm [44]). We choose these two because they are able to take advantage

of the good preconditioners available for this problem, and because they are Krylov-subspace methods—at each step, they compute the best solution in the subspace of *all* previously tried directions (with some approximations, described below) [39,45].

As a test case for the convergence of the algorithms discussed below, we use a 3D diamond lattice of dielectric spheres ($\epsilon = 12$) in air, which has a gap between its second and third bands [3]. Except where otherwise noted, we solve for the first five bands at the L wavevector point with a $16 \times 16 \times 16$ grid resolution in the (affine) primitive cell (~ 22.6 grid points per lattice constant), tracking the error in the eigenvalue trace versus the converged value, with the non-diagonal preconditioner of Eq. (14). A set of five pseudo-random fields are used as the starting point for the eigensolver, and we report the results from the case that converges in the median number of iterations. Note that the number of iterations in practice will often be less than that reported here—not only is the minimum solution tolerance usually bigger, but one typically starts with the fields from a nearby \vec{k} point (yielding a significant “head start” vs. random fields).

3.1 Conjugate-gradient minimization of the Rayleigh quotient

The smallest eigenvalue λ_0 and the corresponding eigenvector y_0 of a Hermitian matrix A are well-known to satisfy a variational problem—they minimize the expression:

$$\lambda_0 = \min \frac{y_0^\dagger A y_0}{y_0^\dagger B y_0}, \quad (15)$$

known as the “Rayleigh quotient,” where † denotes the Hermitian adjoint (conjugate-transpose). One can then compute this eigenpair by performing an unconstrained minimization of the Rayleigh quotient, using a method such as conjugate-gradient [46]. To find the subsequent eigenvalue and eigenvector, the minimization is repeated while maintaining orthogonality to y_0 (through B), and so on; a process known as “deflation.” As in all iterative methods, the matrix A need never be computed explicitly, only the product Ay (and By). This method [39] for computing the smallest eigenpairs of a matrix, also called a “Rayleigh-Ritz” algorithm, was employed by [5] to solve for the eigenstates of Maxwell’s equations in a planewave basis.

3.1.1 The block Rayleigh quotient

We use an extension of the conjugate-gradient Rayleigh-quotient method, adapted from [40], that solves for all of the eigenvectors at once instead of one-by-one with deflation. Such a process is called a “block” algorithm, and it has two advantages in our case: first, it can take advantage of efficient block matrix operations that have superior cache utilization on modern computers [47,48]; second, on parallel machines, block algorithms can communicate in larger chunks that mask latencies. Actually, as described in Sec. 3.4, we employ a hybrid of block algorithms and deflation, but such a generalization can only be made if one has a block method to begin with.

Let Y be an $n \times p$ matrix whose columns are the p eigenvectors with the smallest eigenvalues. Then Y minimizes the trace $\text{tr}[Y^\dagger A Y]$ subject to the orthonormality constraint $Y^\dagger B Y = I$ (where I is the identity matrix) [50]. Although it is possible to perform such a minimization directly by observing the differential geometry of the constraint manifold [49], it is more convenient to introduce a rescaling, similar to that of the Rayleigh quotient, that makes the problem unconstrained.³ If we write:

$$Y = Z (Z^\dagger B Z)^{-1/2} \quad (16)$$

³We also tried using the differential-geometry conjugate-gradient algorithm of [49]. In addition to requiring more matrix operations per iteration than the method described here, it also seemed to work much more poorly with our choice of preconditioners, for unknown reasons.

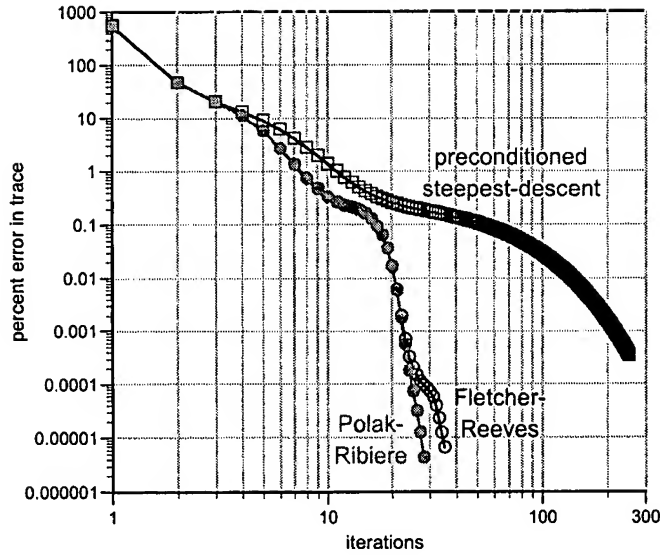


Fig. 2. Eigensolver convergence for two variants of conjugate gradient, Fletcher-Reeves and Polak-Ribiere, along with preconditioned steepest-descent for comparison.

for an arbitrary non-singular $n \times p$ matrix Z , then the orthonormality constraint is automatically satisfied—this is known as the “polar” or “symmetric” orthogonalization of Z , and happens to be optimal from a certain numerical-stability standpoint [51]. More importantly, we can now solve for the eigenvectors by performing an *unconstrained* minimization of the block Rayleigh quotient:

$$\text{tr} [Z^\dagger AZU], \quad (17)$$

where $U = (Z^\dagger BZ)^{-1}$. Once the minimization is completed, the eigenvectors are a superposition of the columns of Z , and can be found by diagonalizing the $p \times p$ matrix $Y^\dagger AY$ where Y is given by Eq. (16).

3.1.2 Conjugate gradient

The conjugate-gradient algorithm is a method for minimizing multidimensional functions by searching along successive “conjugate” directions, chosen so that each line minimization does not spoil the previous ones [46]. Strictly speaking, the algorithm is designed only for quadratic functions, for which it is a Krylov-subspace method. Even non-quadratic forms such as Eq. (17), however, are approximately quadratic near their minima and so conjugate-gradient is beneficial. The minimization direction is usually chosen starting from the steepest-descent (gradient) direction G , which in this case is given by [40]:

$$G = P_\perp AZU, \quad (18)$$

where P_\perp is the projection operator onto the space orthonormal to Z : $P_\perp = 1 - BZUZ^\dagger$. The conjugate minimization direction D is then:

$$D = \hat{K}G + \gamma D_0, \quad (19)$$

where \hat{K} is a preconditioning operation (discussed below), D_0 is the minimization direction from the *previous* step, and γ is either:

$$\gamma = \frac{\text{tr} [G^\dagger \hat{K} G]}{\text{tr} [G_0^\dagger \hat{K} G_0]} \quad (20)$$

in the Fletcher-Reeves variant or:

$$\gamma = \frac{\text{tr} [(G - G_0)^\dagger \hat{K} G]}{\text{tr} [G_0^\dagger \hat{K} G_0]} \quad (21)$$

in the Polak-Ribiere variant, where G_0 is the gradient direction from the previous step. Both variants involve approximations for the derivative of G [49], with Polak-Ribiere being slightly more accurate at the expense of requiring extra storage space for G_0 . If we instead choose $\gamma = 0$, the result is simply the preconditioned steepest-descent algorithm. The relative convergence of these variations is depicted in Fig. 2, and displays a sharp acceleration for conjugate-gradient once the error becomes small, presumably due to the function becoming approximately quadratic. (Our subsequent convergence plots use Polak-Ribiere.)

Once the conjugate direction is determined, one needs to minimize Eq. (17) over $Z' = Z + \alpha D$. By substituting Z' into Eq. (17), one finds a function in terms of α and constant $p \times p$ matrices, which we then minimize via a one-dimensional optimization algorithm specifically designed for use with conjugate-gradient-like methods [52]. Alternatively, one could make a two-point approximation for the second derivative of the function along the line and fit to a quadratic (i.e., apply one step of Newton's method) [38, 40]. This method requires two fewer $O(p^2 n)$ matrix multiplications (out of eight per iteration), but often produces somewhat slower and less reliable convergence.

3.1.3 Preconditioning

Large gains in the rate of convergence can be achieved by choosing a proper Hermitian preconditioning operator \hat{K} in Eq. (19)—ideally, an approximate inverse of the Hessian (second-derivative) matrix of the objective functional, Eq. (17).⁴ One can think of this as an application of the multi-dimensional Newton's method to find the zero of the gradient G : the Newton update $\hat{K}G$ of the solution guess Z is the function G divided by its derivative. Equivalently, if we are at

$$Z = Z^* + \delta Z, \quad (22)$$

where Z^* is the unknown minimum, we wish to solve for $\delta Z \cong \hat{K}G$ to lowest order. Thus, we substitute Eq. (22) into Eq. (18) for the gradient and expand to first order in δZ , noting that $G^* = 0$, and find [53]:

$$G \cong P_\perp (A\delta Z - B\delta Z U Z^\dagger A Z) U, \quad (23)$$

Suppose that Z were rotated to diagonalize the generalized eigenproblem $Z^\dagger A Z x = Z^\dagger B Z x \tilde{\lambda}$. Then, the second term in Eq. (23) would be simply $B\delta Z$ times the current eigenvalue approximations. Inverting Eq. (23) thus involves the inversion of $A - B\tilde{\lambda}$. At this point, however, we make an additional approximation: since the desired eigenvalues λ are small, we neglect that term and use:

$$G \cong P_\perp A \delta Z U. \quad (24)$$

⁴In the literature, "preconditioner" sometimes refers instead to \hat{K}^{-1} , the approximate Hessian.

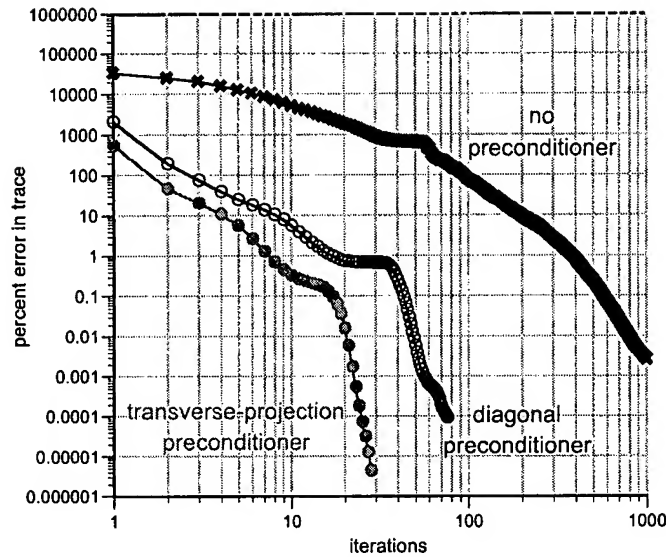


Fig. 3. The effect of two preconditioning schemes from section 2.4, diagonal and transverse-projection (non-diagonal), on the conjugate-gradient method.

An inverse of this equation is given by the Hermitian operation $\delta Z = A^{-1}GU^{-1}$, which is then further approximated by:

$$\delta Z \cong \hat{K}G = \widetilde{A}^{-1}GU^{-1}, \quad (25)$$

where we have used an approximate inverse for A such as those in Sec. 2.4. The effects of the preconditioner of Eq. (25) with our two approximate inverses are demonstrated in Fig. 3, showing significant benefit from the non-diagonal preconditioner.

Additional refinements are possible that we do not consider here. One could solve Eq. (23) exactly by an iterative method, or at least a few iterations thereof to improve the preconditioner. Also, because of the projection operator P_{\perp} , one has a choice of how to invert Eqs. (23, 24)—it has been suggested in a similar context that one should choose δZ in the space orthogonal to Z [43]. This can be done at the expense of a few extra matrix operations, but it did not seem to improve convergence significantly in our case. Such a choice would become more important, however, if one attempted a more accurate preconditioner that inverts expressions of the form $A - B\lambda$, lest singularities arise.

3.2 The Davidson method

In the Davidson method, instead of iteratively improving a single (block) eigenvector approximation Y , one builds up an increasing subspace V that eventually contains the desired eigenvectors [42]. While both the Davidson and conjugate-gradient methods are Krylov-subspace algorithms, this is only exactly true of conjugate-gradient for pure quadratic forms, and of Davidson in the absence of restarting (described below). We summarize the method briefly in the following.

At each iteration, V is an $n \times q$ matrix for some $q \geq p$, whose columns span the current subspace, with $V^{\dagger}BV = I$. The task is to find p new vectors to add to V so that it spans a better approximation for the desired p eigenvectors. To do this, one first computes the best p eigenvectors Y so far: A is projected to the V subspace, $A_v = V^{\dagger}AV$, and the associated eigenvectors are computed for this $q \times q$ matrix; of these eigenvectors, one

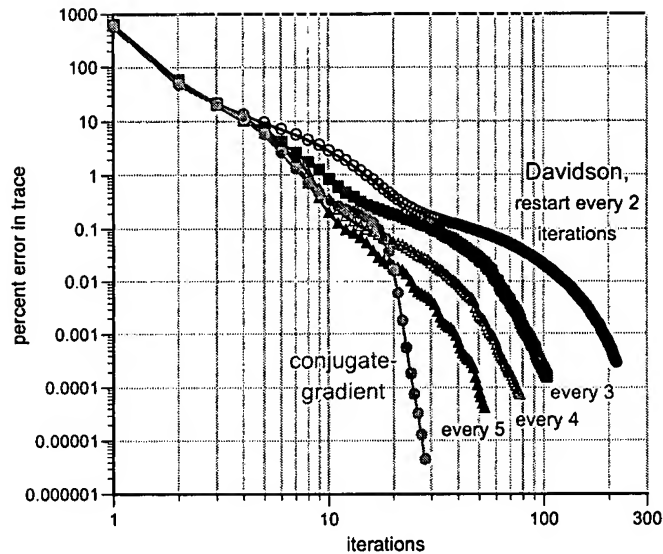


Fig. 4. Comparison of the Davidson method with the block conjugate-gradient algorithm of section 3.1. We reset the Davidson subspace to the best current eigenvectors every 2, 3, 4, or 5 iterations, with a corresponding increase in memory usage and computational costs.

chooses the block of the p smallest eigenvalues, $L = \text{diag}(\lambda_1 \cdots \lambda_p)$, and their associated $q \times p$ eigenvectors Y_v , and computes the “Ritz” eigenvectors $Y = VY_v$. The residual (i.e., error) for this eigenvalue approximation is $R = BYL - AY$. The new directions to be added to V are then given by $D = \hat{K}R$ (orthogonalized, to maintain the orthonormality of V), where \hat{K} is a preconditioning operator.

To keep the subspace limited to a reasonable size, this process is usually restarted every few iterations by setting $V = Y$, with some tradeoff in speed of convergence. (Alternatively, one could keep the r lowest eigenvectors for some $r > p$ [43].) Restarting in this way on *every* iteration (never increasing the subspace) is a form of steepest-descent; a method of this sort was used in [12]. The ideal preconditioner \hat{K} for the Davidson method, remarkably, has been shown to be the same as that of Eq. (23) for trace-minimization, combined with the fact that Y in this case is an orthonormal solution to the projected eigenproblem. That is, $\hat{K}R$ should be an approximate solution for D in [43]:

$$R = P_{\perp} (AD - BDL), \quad (26)$$

where L is the diagonal matrix of eigenvalues from above and P_{\perp} is the projection matrix $1 - BYY^{\dagger}$ as in Sec. 3.1.2. Again, D should ideally be found in the space orthonormal to Y [43]; this is the “Jacobi-Davidson” algorithm. To solve Eq. (26), we make the same approximations as in Sec. 3.1.3 and use $\hat{K}R = \widetilde{A}^{-1}R$. The Davidson method is also adaptable to non-Hermitian operators, and it has thus been employed to treat electromagnetic problems containing absorption [14].

The results of a comparison of the Davidson method described here with the block conjugate-gradient minimization of the Rayleigh quotient from the previous section are shown in Fig. 4. Various restarting frequencies are used, with more infrequent restarts resulting in faster convergence but also higher memory requirements and a greater computational cost per iteration (e.g. more than twice as much memory, and 1.6 times as many floating-point operations for eigenvector matrix products, in our period-5 restarted Davidson compared to conjugate-gradient). Overall, it does not seem to match the

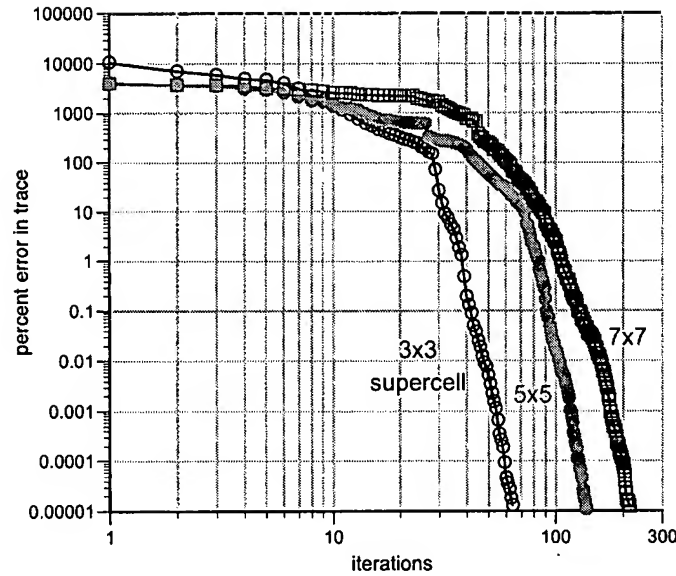


Fig. 5. Conjugate-gradient convergence of the lowest TM eigenvalue for the “interior” eigensolver of Eq. (27), solving for the monopole defect state formed by one vacancy in a 2D square lattice of dielectric rods in air, using three different supercell sizes (3×3 , 5×5 , and 7×7).

performance of the conjugate-gradient method in our case. Nevertheless, we believe it is important that future studies continue to investigate the Davidson method with different preconditioners and restarting schemes, especially since it seems to exhibit more regular convergence than the conjugate-gradient algorithm.

3.3 Interior eigenvalues

One of the most interesting aspects of photonic crystals is the possibility of localized modes associated with point defects (cavities) and line defects (waveguides) in the crystal [1,54]. In this case, the desired modes lie in a known frequency range (the band gap) in the *interior* of the spectrum. Moreover, since a supercell of some sort must be used to surround the defect with sufficient bulk crystal, the underlying non-localized modes are “folded” many times (increasing their number by the number of primitive cells in the supercell). Ideally, one would like to compute only the defect modes in the band gap, without the waste of computation and memory of finding all the folded modes below them. One way to accomplish this is to use the FDTD algorithms referenced in Sec. 1. Here, we demonstrate that it is also possible to compute only the interior eigenvalues using iterative frequency-domain methods.

The center of the band gap is known, so we can state the problem as one of finding the eigenvalues and eigenvectors closest to the mid-gap frequency ω_m . Since the methods presented above compute the minimum eigenvalues of an operator, we can apply them directly merely by shifting the spectrum [55]:

$$\hat{A}'_{\vec{k}} = \left(\hat{A}_{\vec{k}} - \frac{\omega_m^2}{c^2} \right)^2. \quad (27)$$

This operator has the same eigenvectors as $\hat{A}_{\vec{k}}$, but its lowest eigenvalues are the ones closest to ω_m , and thus we can compute a single defect state without computing any other eigenstates. As a preconditioner for this operator, we again approximate ω_m as

small and simply use $\hat{K}G = \widetilde{A}^{-2}GU^{-1}$, essentially the square of Eq. (25). This imperfect preconditioner and the worsened condition number of Eq. (27) will lead to slower eigensolver convergence, but this is generally more than compensated for by the reduction in the number of bands computed (and the resulting savings in memory and time per iteration). To demonstrate this “interior” eigensolver, we consider the case of a two-dimensional square lattice of dielectric rods in air, with a single rod removed from the crystal. Such a defect supports one confined TM mode in the gap [54], and in Fig. 5 we plot the eigensolver convergence of the lowest TM eigenvalue of Eq. (27), for 3×3 to 7×7 supercells with a resolution of 16 grid points per lattice constant. (To find the same state in a 7×7 supercell via the ordinary method, $p = 49$ bands must be computed.)

An alternative method for computing interior eigenvalues using a modified version of the Davidson method has also been suggested [43] and we intend to investigate this algorithm in future work, along with more accurate preconditioners (e.g. based on iterative linear solvers).

3.4 To block or not to block?

We have described the use of block eigensolvers to solve for all of the desired eigenstates simultaneously. The classical algorithm of computing them one by one with deflation has its advantages, however. First, it uses less memory— $O(pn)$ memory is required to store the eigenstates themselves, but only $O(n)$ memory is required for auxiliary matrices such as G and D , versus $O(pn)$ for the block method. (At least two such auxiliary matrices seem to be required for conjugate gradient, although three is more convenient, and one more for the Polak-Ribiere method.) Second, because our preconditioner is a better approximation for the lower bands (where ω is smallest), the eigensolver convergence of the upper bands is slower—it is more efficient to compute them separately. Third, the eigensolver iterations themselves can require fewer operations for the deflation algorithm, as described below.

Since blocked algorithms maintain inherent cache-reuse advantages on modern computer architectures [47], however, it makes sense to consider a hybrid approach: we use the blocked eigensolver to solve for b bands at a time, with deflation to orthogonalize against the previous bands. The optimal choice of b depends upon the particular problem size considered and hardware/software details, but it typically ranges between five and fifteen bands. (As long as b is several times smaller than p , the memory advantages of deflation are realized.) Often, a factor of two or more in speed is gained compared to either $b = 1$ or $b = p$.

Let us consider the arithmetic complexity of the iterations. Each eigensolver iteration for b bands requires $O(\ell nb^2)$ operations for ℓ matrix operations such as ZU , and $O(bn \log n)$ operations for A in a planewave basis. Furthermore, orthogonalization against the q previous bands requires $O(2nbq)$ operations (once or twice per iteration, to maintain numerical stability). Assuming that all bands require the same number of iterations, this must be repeated p/b times. Thus, each eigensolver iteration requires a total of $O(\ell npb) + O(pn \log n)$ operations for the bands themselves, plus $O(np(p - b))$ operations for the orthogonalization. So, if ℓ is roughly greater than the number of orthogonalizations per iteration ($\ell \simeq 8$ in our implementation), then deflation requires fewer operations per iteration than computing all of the bands at once. Of course, the actual running time scales in a more complicated way, due to the cache, CPU pipeline, and other issues.

3.5 Scaling

One final question that we wish to address is that of scaling: how does the eigensolver convergence rate scale with the size of the problem? Specifically, we consider the conver-

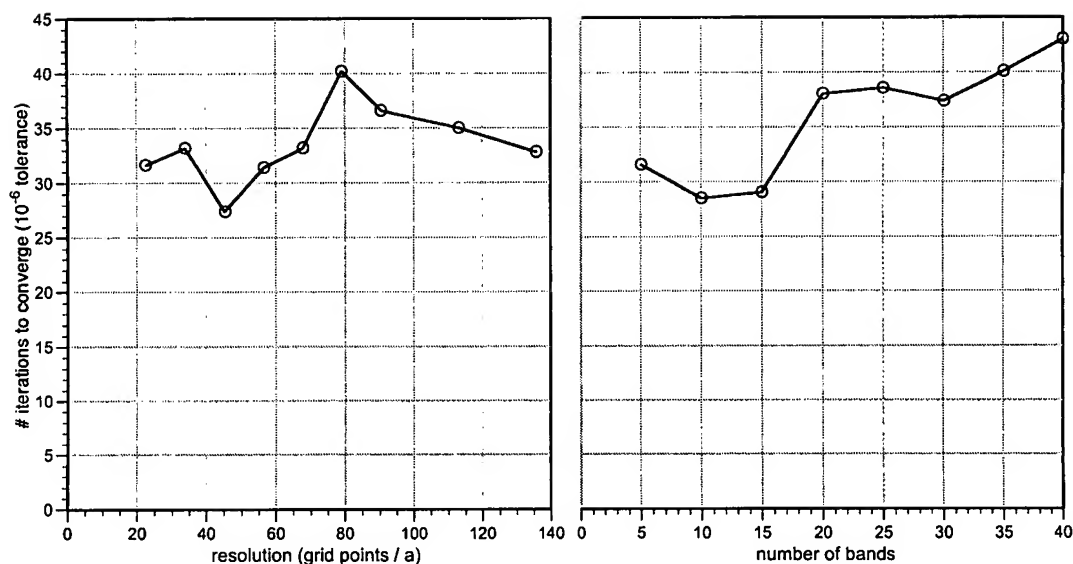


Fig. 6. Scaling of the number of conjugate-gradient iterations required for convergence (to a fractional tolerance of 10^{-7}) as a function of the spatial resolution (in grid points per lattice constant, with a corresponding planewave cutoff), or the number p of bands computed.

gence of the block conjugate-gradient algorithm for the diamond structure as we increase either the resolution or the number of bands. In both cases, we plot the mean number of iterations (over five runs with random starting fields) for the eigensolver to converge to within a fractional tolerance of 10^{-7} (which is probably lower than the tolerance that would be used in practice). In the case of the increased number of bands, we solve for the bands in blocks of $b = 5$ bands, as described in Sec. 3.4, and use the average number of iterations per block. The results, plotted in Fig. 6, demonstrate that the number of iterations increases only very slowly as the problem size is increased. (We suspect that the slowed convergence for the increased number of bands results largely from the worsening of the small- ω approximation in our preconditioner.) In contrast, FDTD algorithms must scale their number of iterations (or, strictly speaking, Δt) linearly with the spatial resolution in order to maintain stability.

4 Conclusion

We have presented efficient preconditioned block-iterative algorithms for computing eigenstates of Maxwell's equations for periodic dielectric systems using a planewave basis. Such methods, combined with appropriate effective dielectric tensors for accuracy, interior eigensolvers to compute only the desired modes in defect systems, and a flexible and freely-available implementation, provide an attractive way to perform eigen-analyses of diverse electromagnetic systems.

Acknowledgments

This work was supported in part by the Materials Research Science and Engineering Center program of the National Science Foundation under Grant No. DMR-9400334 and by the U. S. Army Research Office under Grant No. DAAG55-97-1-0366. S. G. Johnson is grateful for the support of a National Defense Science and Engineering Graduate Fellowship and an MIT Karl Taylor Compton Fellowship.